International Journal of Mechanical Engineering and Technology (IJMET)
Volume 10, Issue 1, January 2019, pp. 335–343, Article ID: IJMET_10_01_034
Available online at http://www.iaeme.com/ijmet/issues.asp?JType=IJMET&VType=10&IType=1
ISSN Print: 0976-6340 and ISSN Online: 0976-6359

ATTITUDE QUADROTOR CONTROL SYSTEM WITH OPTIMIZATION OF PID PARAMETERS BASED ON FAST GENETIC ALGORITHM

Nasra Pratama Putra, Gerzon Jokomen Maulany
Department of Information Systems, Faculty of Engineering,
UniversitasMusamus, Merauke, Indonesia

Frans Xaverius Manggau
Informatics Engineering Department, Faculty of Engineering,
UniversitasMusamus, Merauke, Indonesia

Philipus Betaubun
Civil Engineering Department, Faculty of Engineering,
UniversitasMusamus, Merauke, Indonesia

ABSTRACT

Quadrotor control is needed so that the quadrotor can float close to the stationary state. For that we need control techniques. One control technique that can be designed and implemented in quadrotor is PID control. PID parameter tuning using the Genetic Algorithm technique can speed up the manual tuning process. The weakness in the application of the Genetic Algorithm rule is that it often rejects important information found in other individuals and causes premature convergence, especially at the beginning of the generation. These problems can be overcome by using crossover and mutation rules with different probability levels according to fitness values and evolutionary processes. The results of the study using fast genetic algorithm techniques obtained constants Kp, Ki and Kd with the lowest rise time and overshoot, namely 0.010, 0.001 and 0.036 at the pitch angle. At the roll angle, they are 0.010, 0.001 and 0.03. At yaw angle 0.018, 0.006 and 0.043. Comparison of PID tuning simulations using fast genetic algorithm with genetic algorithm standards, shows that fast genetic algorithm has increased optimum generation achievement faster by 26.67% at pitch angle, 44% at roll angle and 20% at yaw angle. This condition has an effect on increasing simulation execution time, where fast genetic algorithm is 26.4% faster at pitch angle, 38.05% at roll angle, and 24.19% at yaw angle.

Keywords: PID, quadrotor, genetic, algorithm.

Cite this Article: Nasra Pratama Putra, Gerzon Jokomen Maulany, Frans Xaverius Manggau and Philipus Betaubun, Attitude Quadrotor Control System with Optimization of PID Parameters Based On Fast Genetic Algorithm, International Journal of Mechanical Engineering and Technology, 10(1), 2019, pp. 335–343.
http://www.iaeme.com/ijmet/issues.asp?JType=IJMET&VType=10&IType=1

INTRODUCTION

Quadrotor or often called quadcopter is a type of Unmanned Aerial Vehicle (UAV). Quadrotor can fly in the air using the help of four motors and four propellers (Dharmawan and Putera, 2012). By using the thrust of the rotation of each motor, it can produce movement on the quadrotor. The control system that regulates this is the attitude control system. The attitude control system can be generated through system modeling. At present there are two modeling methods that can be used for attitude from quadrotor dynamics, namely Lagrange equations and Newton II laws (Dharmawan et al., 2014). As for the well-known control system and can be implemented directly on quadrotor, it is a PID control (Bouabdallah et al., 2004).

In the PID control there are 3 gain pieces that work on errors, namely Kp, Ki, and Kd. These three values greatly influence the optimal level of PID control (Priyambodo et al., 2004). Tuning Kp, Ki, and Kd can be done manually to achieve the appropriate system response. The traditional method that is popularly used in PID is Ziegler Nichol. The weakness of the Ziegler Nichol method is that it often does not provide optimal tuning. Furthermore, this method easily generates high overshoot waves. Besides using the Ziegler Nichol method, there are also methods that use the intelligence approach. This approach uses optimization methods based on natural evolution.

Evolutionary computation is a method that from the very beginning of its creation mimicked Charles Darwin's theory of evolution, "Survival of the fittest" (Priyambodo et al., 2004). One method that applies optimization techniques based on natural evolution is through the process of mutation, crossovers and selection is Genetic algorithm (Sivanandam and Deepa, 2008). The unknown parameter value for PID is definitely coded as a chromosome in binary code. The binary code results are then evaluated using objective functions based on costs in time domain (Jones and Oliveira, 1995). There are also several studies that implement genetic algorithm methods in quadrotor PID control systems (Abdollahi et al., 2015; Bolandi et al., 2013; Li et al., 2014). But the disadvantage of applying the Genetic Algorithm rule is that it can reject important information contained in other individuals and can cause premature convergence.

Premature convergence is a situation where Genetic Algorithm often cannot get a global optimal solution but is trapped into a local optimal solution. This situation is also supported by the application of static probability values on the crossover rate and mutation rate. Based on the problem, this study tries to apply different rules for determining the crossover rate and mutation rate to improve the search process for the values of Kp, Ki, and Kd which are optimal for quadrotor transfer functions. Determination of crossover rate and mutation rate is adjusted to the fitness value of each individual. This condition can cause the probability of crossover rate and mutation rate to decrease according to increasing generation. The decrease in crossover rate and mutation rate is applied when it is found that the individual fitness value is higher than the average individual fitness in previous evolution. Crossover rate and mutation rate will return static value if it is found that individual fitness values are lower or equal to the average individual fitness in previous evolution.

METHODOLOGY

The method used in tuning PID in this study is to use the Genetic Algorithm that has been modified so that it is not trapped into a local optimal solution or called Fast Genetic Algorithm. To be able to apply this method, then do the various initial stages to get the data needed, namely:

Modeling Analysis

Newton-Euler approach to determine the dynamics of quadrotor, namely through equations (1) (2) (3) (4):

$$\dot{\xi} = v \tag{1}$$

$$m\dot{v} = f \tag{2}$$

$$\dot{R} = R\widehat{\Omega} \tag{3}$$

$$I\dot{\Omega} = -\Omega \text{x}\, I\Omega + \tau \tag{4}$$

The quadrotor inertia moment is described in the equation (5)(6)(7):

$$I_{xx} = \frac{mr^2}{4} + \frac{mh^2}{6} + 2mr^2 + \frac{MR^2}{4} + \frac{MH^2}{12} \tag{5}$$

$$I_{yy} = \frac{mr^2}{4} + \frac{mh^2}{6} + 2mr^2 + \frac{MR^2}{4} + \frac{MH^2}{12} \tag{6}$$

$$I_{zz} = \frac{MR^2}{2} + 4mr^2 \tag{7}$$

Where M is the quadrotor middle box mass, m is the brushless motor mass, R is the distance between the ends of the middle part of the quadrotor to the center of mass, r is the brushless motor radius, h is the brushless motor height, H is the height of the quadrotor center, and l is the length between the ends of the quadrotor to the center of mass. The final equation for angular acceleration $\ddot{\phi}$, $\ddot{\theta}$, and $\ddot{\psi}$ is linearized based on the drift state into the equation (8)(9)(10):

$$\frac{\phi(s)}{F(s)} = \frac{l}{I_{xx}s^2} \tag{8}$$

$$\frac{\theta(s)}{F(s)} = \frac{l}{I_{yy}s^2} \tag{9}$$

$$\frac{\psi(s)}{F(s)} = \frac{l}{I_{zz}s^2} \tag{10}$$

Transfer function

The block diagram of Laplace's domain transfer function pitch angle, roll angle, and yaw angle in the quadrotor is in accordance with Figure 1.

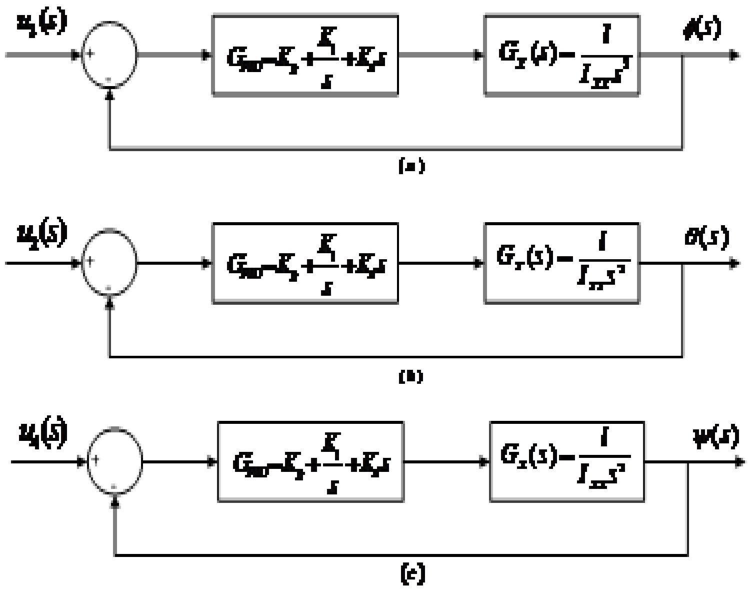

Figure 1. The PID Controller Inserted Block Diagram: (a) Roll Angle, (b) Pitch Angle, (c) Yaw Angle

Fast Genetic Simulation Algorithm

In Fast Genetic Algorithm, the crossover (Pc) probability value decreases with increasing generation. This causes the population size in each generation to decline. But it can speed up the computing process. Equation (11) describes the determination of the probability value *crossover* (Pc).

$$P_c = P_{c1} - ((m-1) * P_{c1}/M) \tag{11}$$

Where:

P_{c1} = The probability of a specified crossover is 1

m = Generation evolution

M = Total Generation

The mutation process that will be used in this study is a mutation for binary numbers. A simple way to do binary mutations is to replace one or more gene values from chromosomes. The mutation method is known as Flip Bit. The mutation process also depends on the probability of mutation (Pm). In Fast Genetic Algorithm, the probability value of mutation (Pm) decreases every generation. The goal is also to speed up the computing process. Equation (12) describes the determination of the mutation probability value (Pm):

$$P_m = P_{m1} - ((m-1) * P_{m1}/M) \tag{12}$$

Where:

P_{m1} = The mutation probability is set to 0.01

m = Generation evolution

M = Total Generation

Overall, the PID optimization simulation using Fast Genetic Algorithm is shown in Figure 2.

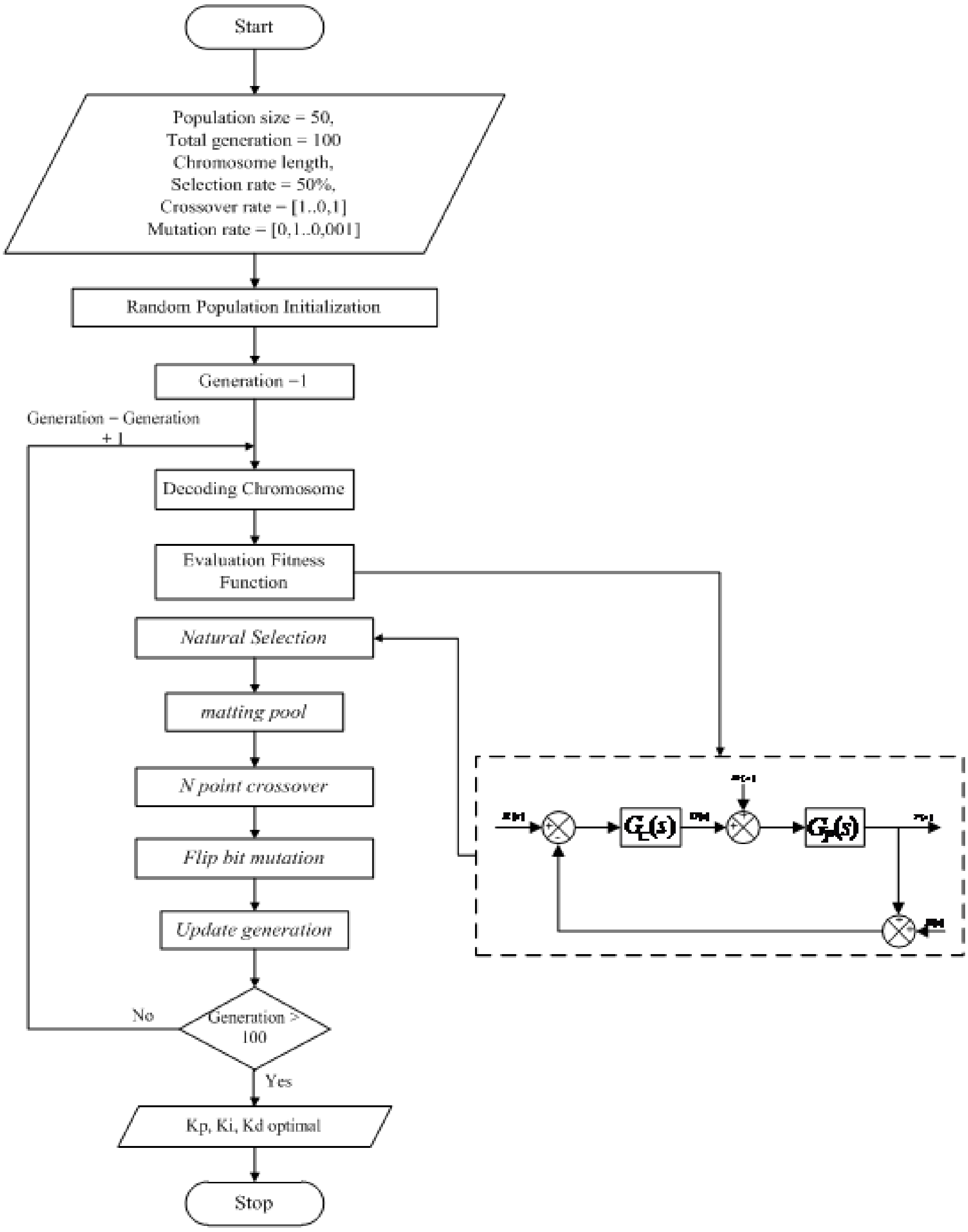

Figure 2. Simulation of PID Optimization with Fast Genetic Algorithm

In addition, other useful methods were provided (Istanto and Manggau, 2018; Lamalewa and Maulany, 2018; Latuheru and Sahupala, 2018; Waremra and Bahri, 2018).

RESULTS

Simulation Implementation

The difference between ordinary Genetic Algorithm and Fast Genetic Algorithm lies in the application of crossover and mutation probabilities. In Genetic Algorithm, the probability of crossover and mutation is set statically. In Fast Genetic Algorithm, the crossover and mutation probability is adjusted differently for each generation. The simulation code is described in Figure 3 and Figure 4.

```
%CROSSOVER
Nmate=size(mpoll);
random=rand(1,Nmate(1));
parentInd=[];
j=1;
for i=1:Nmate(1)
if random(i) < co_rate(generasi)
parentInd (j)=i;
j=j+1;
end;
end
% pairs chromosomes and performs crossover
for ic=1:2:length(parentInd)
if ic>=length(parentInd)
break;
else
Nmate=size(mpoll);
cross=ceil((L-1)*rand());
mpoll(Nmate(1)+1,1:cross)=mpoll(parentInd (ic),1:cross);
mpoll(Nmate(1)+1,cross+1:L)=mpoll(parentInd (ic+1),cross+1:L);
mpoll(Nmate(1)+2,1:cross)=mpoll(parentInd (ic+1),1:cross);
mpoll(Nmate(1)+2,cross+1:L)= poll(parentInd (ic),cross+1:L);
end
end
```

Figure 3. Code of the Fast Genetic Algorithm Crossover Program

```
%MUTATION
mutrate=0.001+([0:1:n_iter]*(0.01/n_iter));
Nmuts=ceil(mutrate*(Npop-1)*L);
    Nmate=size(mpoll);
for ic=1:Nmuts(generasi)
        ix=ceil(Nmate(1)*rand);
        iy=ceil(L*rand);
        mpoll(ix,iy)=1-mpoll(ix,iy);
end%ic
```

Figure 4. Mutation Program Code Fast Genetic Algorithm Simulation Test

Simulation Test

When Kp is 0.0869, Ki is 0.001 and Kd of 0.0196 has a maximum peak value of 2.9 degrees. This means that the quadrotor will be stable to float while still being able to keep the tilt angle no more than 2.9 degrees. The value of settling time obtained in these conditions is 2.59 seconds. This means that the angle of slope will decrease until it reaches stationary state with a travel time of around 2.59 seconds. The number of generations to achieve this condition is around 34 generations. The simulation execution time for this condition is 84.7093 seconds. Impulse response pitch and roll angles along with progress tuning for each generation are shown in Figure 5.

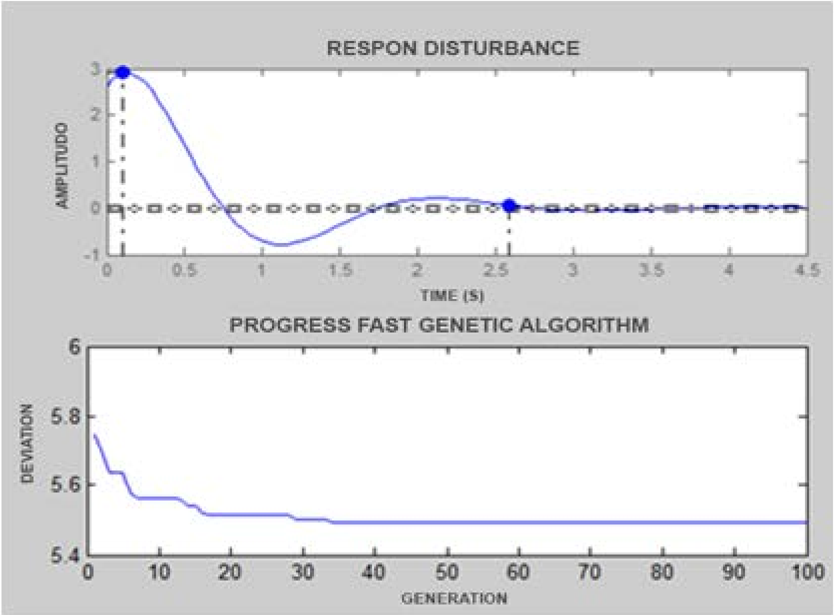

Figure 5. Results of pitch and roll angle simulations with Fast Genetic Algorithm

Comparative data on the results of achieving minimum cost in tuning Genetic Algorithm and Fast Genetic Algorithm simulations are shown in Table 1.

Table 1. Comparison of search results minimum cost

Algorithm	Control constant			Cost	Achievement of optimum generation	Execution time (seconds)
	Kp	Ki	Kd			
FGA	0.0869	0.001	0.0196	5.4941	34	84.7093
SGA	0.0948	0.0126	0.0289	5.4938	99	150.068

In Table 1 it is known that there is almost no difference between the minimum cost generated through the PID tuning simulation using both the Genetic Algorithm and Fast Genetic Algorithm. But there is an increase in the achievement of optimum generation by Fast Genetic Algorithm, which is the generation when the minimum cost is found:

$$abs\left(\frac{34-99}{99}\right)\times\ 100\% = 65.66\%$$

There was also an increase in simulation execution time between tuning simulations using the Standard Genetic Algorithm and Fast Genetic Algorithm, where the use of the Fast Genetic Algorithm method was faster than:

$$abs\left(\frac{84{,}7093-150{,}068}{150{,}068}\right)\times\ 100\% = 43.55\%$$

Dynamic Testing

The test results are carried out by flying quadrotor at a certain height with a pwm value of more than 1500. The PID constant used is the PID constant tuning using the Fast Genetic Algorithm method. The response you want to know is the peak value and settling time.

The pitch and roll angle response uses the constants Kp, Ki and Kd obtained from the simulation using Fast Genetic Algorithm shown in Figure 6 and Figure 7.

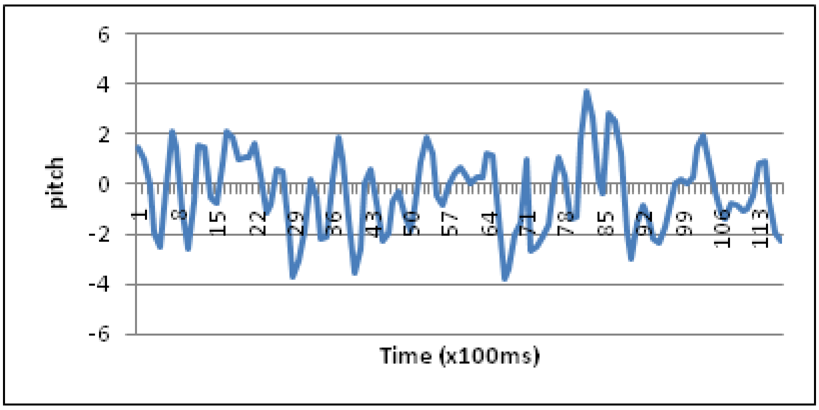

Figure 6 Response of dynamic test pitch angle with PID results of tuning Fast Genetic Algorithm

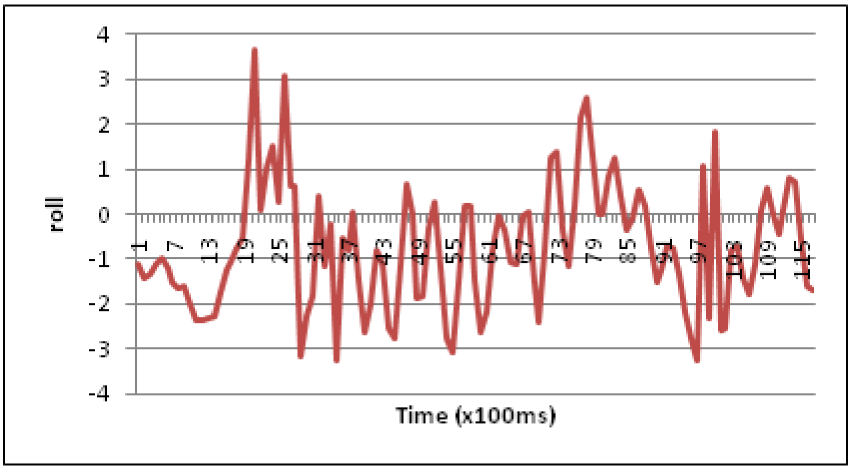

Figure 7. Dynamic roll angle test response with PID as a result of tuning Fast Genetic Algorithm

In Figure 6 and Figure 7, the desired pitch and roll angle is 0 °. The peak value of the dynamic test results indicate that the quadrotor has been able to fly by keeping the pitch and roll angle of less than 4 degrees. These results differ slightly from the results of the simulation response which state the highest slope angle of 2.9 degrees.

CONCLUSIONS

The conclusions that can be drawn from the results of this study are as follows:

- The results of comparison of PID tuning simulation using fast Genetic Algorithm compared to using the Standard Genetic Algorithm at pitch and roll angles, shows that Fast Genetic Algorithm has increased attainment of a minimum cost faster 65.66%. While Fast Genetic Algorithm simulation execution time increased 43.55% faster than the Standard Genetic Algorithm.
- The peak value or slope of the quadrotor angle obtained from the dynamic test results with the simulation experience different values, but relatively small.

REFERENCES

[1] Abdollahi, A., Tabar, A.F., dan Khodadadi, H., 2015. Optimal Controller Design for Quadrotor by Genetic Algorithm with the Aim of Optimizing the Response and Control Input signals. *Cumhuriyet Science Journal*, **36**: 135–147.

[2] Bolandi, H., Rezaei, M., Mohsenipour, R., Nemati, H., dan Smailzadeh, S.M., 2013. Attitude Control of a Quadrotor with Optimized PID Controller. *Intelligent Control and Automation*, **04**: 335

[3] Bouabdallah, S., Noth, A., dan Siegwart, R., 2004. 'PID vs LQ control techniques applied to an indoor micro quadrotor', dalam: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings*. Dipresentasikan pada 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings, hal. 2451–2456 vol.3.

[4] Dharmawan, A. dan Putera, C.A.L., 2012. Purwarupa Sistem Integrasi Quadcopter dan Mobile Robot. *IJEIS - Indonesian Journal of Electronics and Instrumentation Systems*, **2**: 97–108.

[5] Dharmawan, A., Simanungkalit, Y.Y., dan Megawati, N.Y., 2014. Pemodelan Sistem Kendali PIDpada Quadcopterdengan Metode Euler Lagrange. *IJEIS - Indonesian Journal of Electronics and Instrumentation Systems*, **4**: 13–24.

[6] Istanto, T and Manggau, FX. 2018. Analysis of the Power of Wifi Signals on the Informatics Engineering Laboratory of Musamus University Using Insidder, International Journal of Mechanical Engineering and Technology 9(13), pp.266–272.

[7] Jones, A.H. dan de Moura Oliveira, P.B., 1995. 'Genetic auto-tuning of PID controllers', dalam: *Genetic Algorithms in Engineering Systems: Innovations and Applications, 1995. GALESIA. First International Conference on (Conf. Publ. No. 414)*. Dipresentasikan pada Genetic Algorithms in Engineering Systems: Innovations and Applications, 1995.

[8] Lamalewa, L. and Maulany, GJ. 2018. Application of Case Based Reasoning and Nearest Neighbor Algorithm for Positioning Football Players, International Journal of Mechanical Engineering and Technology 9(13), pp.258–265.

[9] Latuheru, RD and Sahupala, P. 2018. Design of Methane Gas Capture Installation from Garbage Waste, International Journal of Mechanical Engineering and Technology, 9(10), pp. 279–287.

[10] Li, X., Qu, S., dan Cai, L., 2014. Studing of PID Parameter Optimization Method for Attitude Controller of Quadrotor. *Journal of Computational Information Systems*, **10**: 7805–7812.

[11] Priyambodo, T.K., Putra, A.E., dan Dharmawan, A., 2015. 'Optimizing control based on ant colony logic for Quadrotor stabilization', dalam: *2015 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*. Dipresentasikan pada 2015 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES), hal. 1–4.

[12] Sivanandam, S.N. dan Deepa, S.N., 2008. *Introduction to Genetic Algorithms*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[13] Waremra, RS and Bahri, S. 2018. Identification of Light Spectrum, Bias Index and Wavelength in Hydrogen Lights and Helium Lights Using a Spectrometer, International Journal of Mechanical Engineering and Technology 9(10), pp. 72–76.